Team A4 Status Report

Subtask A: Finished

2D graphical representations and animations of all characters were achieved and finished making used of the MVC design patter. By creating for each model an entity wrapper in the original world library, more functions such as entity status changing can be incorporated into functions that eventually enabled corresponding behaviors animation. Status includes IDLE, ATTACK, MOVING, HURT, DYING. In this case, the protagonist action can be visualized in the 2D view through animation, same as for the enemies regardless the type.

Healthpack visualization will be based on the health values that a different representation is selected.

Protagonist movement is realized by key press: W for up, A for left, S for down, and D for right.

For tile animations especially, an Observer pattern was implemented to avoid unnecessary waste of animation resources that only when the tiles are poisoned or thundered will there be animations on the tile.

A scaling function was implemented to decide the graphical representation of the items dimensions other than the logical tile grid dimensions.

The protagonist can move into the four directions, tiles are associated with an image depending on their value, some tiles consume more energy when you move onto them.

Autoplay can be used to reach the next level, the used path will be visualized.

The level of difficulty, easy, medium, hard, can be set by the user, resulting in an increasing p- and x- enemy ratio as well as enemy numbers with lessened healthpack quantities.

For any tile index calculation, a similar algorithm inspired by pathfinder was implemented that returns (y coordinate • number of columns + x coordinate) for direct indexing, improving efficiency.

To ensure that the protagonist can survive the game, the healthpack will not vanish but teleport to other locations for the protagonist to use in the future.

Subtask B: Finished

Switching between graphical views and text views are done via clicking on respective tabs. For Text visualization, QTextEdit was implemented and the rich text preperty was employed that the text visualization was realized in the format of HTML to adapt the color changes requirement.

A commented-out realization without coloration use ASCII art-like, string-based visualization plaintext, which visual effect is identical to the result shown in lab guidelines PDF.

For each status (IDLE, ATTACK, MOVING, HURT, DYING) of any entity (protagonist, enemy, penemy, xenemy, affected, no-longer-affected, and planned-path tiles,), there is a color assigned to it that based on the status of the entity the visualized color changes.

User may interact with the game in text view via conventional keypress (WASD) or through the textual message widget that is set focused. The widget will act as console that process commands included in the command map. The current command map includes up, down, right left, goto x y (triggers pathfinder), attack nearest enemy, take nearest healthpack, and help (this prints a list of all available commands).

Subtask C: Finished

The overlay has the world map set as the background with z level of minus one, that on top of which paints the additional tile graphical representation based on the tile value as the data layer, which graphical item dimensions adhere to the logical dimension. The tile values will determine the tile graphic index of which, which frames has a size of 14.

For tiles with value larger than one to infinite, a reddish - tile representation was implemented that contrast the rest green-to-blue calm colors, indicating high energy consumption that the protagonist will be dead immediately due to out of energy.

For the portals, planned-path (triggered by pathfinder), and affected tiles, similar concepts were implemented that rather based on the Z values, the sequence of painting property was also utilized to manage the graphic representations layers.

A correlation of graphical scaling is implemented in the zooming in and out functions and are to be centered around the protagonist. A clipping to zooming ration is also implemented.

Subtask D: Finished

Aside from difficulty, five world levels are initially created among different world models to serve as the data caching without utilizing the caching functions or QCache available for graphical caching provided by Qt, which can be a future improvement.

The protagonist can move to special portal (exit or start) which will result in a different map being loaded. The level switch is implemented as two-ways that protagonist reaching the exit portal will transport the protagonist to the new world model, while going back to the start portal will move the protagonist back to the previous world model.

The world models are stored in a vector in a sequential way and is accessed with index. This loading of new worlds is be done in a short time.

Subtask E: Finished

User interfact

Visualization of protagonist health and energy bars are implemented in the mainwindow and employs a signal-and-slot mechanism that allows immediate reflections of status.

Graphical and text-based representations of the world status are synchronized and can be dynamically switched by clicking respective tabs.

Input commands in the text-based representations are implemtned and described in Subtask B.

Zoom in/out and related scaling functions are implemented in graphical view. It is noteworthy that when zooming, the view scrolling bar are still intact and can be used.

Xenemy

Methods deciding the number of xenemy nunbers were inspired by how penenmy is created in the world library, including concepts of ratio decision, used index, and at-least-one-xenemy rule.

The behavior of xenemy includes random initiated health, able to attack by thundering surrounding tiles which thundered radius correlates to xenemy's strength. The xenemy will teleports after each hit.

It is worth noting that for both poison and thundered tiles, the strength of the tiles will eventually die that if the protagonist steps on which after the tiles dies he/she will not be affected. Yet, if the protagonist steps onto the affected tiles while it is still active, the protagonist can easily die.

Future improvement, to add onto the excitement level of the game, a protagonist-chasing enemy can be employed, or simply transform the xenemy to chase the protagonist from a certain onset or after being attacked since current game is rather unexcited.